



In der Zwickmühle

Also ich weiß ja nicht, ob es so eine gute Idee wäre, meinen Quellcode dafür zu verwenden.

Wahrscheinlich würde er die Qualität der künstlichen Intelligenz nicht gerade erhöhen. Aber damit Sie mir folgen können, muss ich wohl von vorne anfangen. Microsoft. Ja, Microsoft hat nicht nur GitHub gekauft, sondern nun auch ein Tool vorgestellt, das GitHub Copilot heißt. Das soll den Entwicklern da draußen das Coden vereinfachen, indem das Tool Codefragmente für die nächsten Zeilen vorschlägt, bevor Sie überhaupt nur in die Nähe der Tastatur kommen.

Woher weiß es aber, welche Zeilen es vorschlagen soll, und wo zum Henker kommt der Code dann her? Hinter Copilot steckt eine KI, und diese muss selbstverständlich trainiert werden. Und womit bitte schön kann man so eine KI trainieren? Richtig. Mit Quellcode, oder besser: Massen an Quellcode. Und zufällig hat Microsoft Massen an Quellcode da.

Genau: Der Windows-Quellcode ist es nicht. Denn der ist nicht öffentlich verfügbar.

Microsoft hat stattdessen dafür Quellcode aus den öffentlichen GitHub-Repositories verwendet. Nur hätte sich Microsoft damit – so der Vorwurf [1] – fremden geistigen Eigentums bedient.

Aber es wird noch besser: Wenn Copilot dann ein Codefragment vorschlägt, könnte es ja sein, dass dieses Fragment originär von einem GitHub-Nutzer stammt: Der Codeschnipsel hat es durch die KI bis zum Nutzer von Copilot geschafft. Man verwendet also unter Umständen Code, den jemand geschrieben hat, der nicht weiß, dass man ihn verwendet.

Ist das nun wieder so eine typisch deutsche Ansicht? Ich bin, also seh ich auch Probleme überall. Oder sind die Einwände berechtigt? Treiben wir das Ganze auf die Spitze: Ihre Firma kündigt Ihnen, weil eine KI, die mit Ihrem Quellcode trainiert wurde, überraschenderweise genauso guten Code schreibt wie Sie.

Allerdings ist die KI wesentlich günstiger als Sie. Sie wird nicht krank, ist nie schlecht gelaunt und fordert keinen Kaffeevollautomaten. Wie können Sie dem entgegen? Schreiben Sie einfach miesen Code. Den will dann keiner für das Training einer KI benutzen. Allerdings will Ihr Arbeitgeber den Code wahrscheinlich auch nicht verwenden und sieht sich lieber nach einem anderen Entwickler um.

Viel Spaß mit der dotnetpro

Tilman Börner
Chefredakteur dotnetpro

PS: Auf der Editorial-Seite der dotnetpro 8/2021 stand, dass Blazor Desktop eine WebAssembly-Technologie sei. Das ist nicht korrekt. Bei Blazor Desktop ist der Host eine .NET-basierte Desktop-Anwendung.

[1] Built on Stolen Data, www.dotnetpro.de/SL2109Edi1



Tobias Richling

zeigt Ihnen, wie Sie eine Webapplikation in einen Container packen (S. 8)



Stefan Papp

baut Infrastruktur für die Entwicklung mithilfe von Jenkins und Kubernetes auf (S. 30)



Bernhard Pichler

hat in einem Artikel die Feinheiten von Exceptions zusammengefasst (S. 98)